Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift

批量归一化：加速深度网络训练

翻译：Jenny(初)，管枫(复)，任远航(审)

绪论：

在深度神经网络的训练过程中，先前层参数的调整会导致之后每一层输入值的分布发生变化，这种现象使模型的训练变得很复杂。所以在深度神经网络模型的训练中，通常需要仔细选取初始参数并采取较小的学习率，这不但导致模型训练的效率低下，而且使得饱和非线性模型的训练极为困难。我们把这种现象称为内部协变量转移(covariate shift)，并通过归一化(normalizing)每层的输入来解决这个问题。我们方法的强大之处在于把归一化的步骤作为模型训练架构的一部分来实现, 并且对每个训练小批量都执行归一化。批量归一化允许我们使用很高的学习率并且对初始化不太在意。它在一定情况下也可以起到正则化的作用，并减轻了对Dropout的需求。我们在最先进的图像分类模型中使用批量归一化法，在减少了14倍训练步骤的情况下实现了与原模型相同的精度，并以显著增量击败了原始模型。我们使用批量归一化的网络模型，增强了在ImageNet分类上发布的最佳结果:获得了4.9%前5验证误差（和4.8%测试误差），这超出了人类评估者的准确率。

简介：

深度学习极大地提升了视觉，语言和许多其他领域的最先进技术。随机梯度下降已经被证明是训练神经网络一个有效的方法，并且随机梯度下降变量如动量和Adagrad已经被用来去获得最先进的性能。随机梯度下降优化网络的参数，以便最小化损失



其中是训练集。训练按步骤进行，在每一步我们考虑一个大小为的小批量。这个小批量被用来去近似相关参数的损失函数的梯度，通过计算



使用小批量的样本，而不是一次一个样本，在几个方面有有帮助的。首先，小批量上的损失梯度是训练集上梯度的一个估计，其质量随着批量大小的增加而提高。第二，由于现代计算平台提供的并行性，对一个批量的计算比各个样本的个计算更有效。

当随机梯度是简单有效的，它需要仔细调整模型超参数，特别是使用在优化中的学习率以及模型参数的初始值。由于每层的输入受所有先前层的参数影响的事实，使训练复杂化，以致于网络参数的小变化随着网络变得更深而放大。

由于层需要连续的适应新的分布，层输入的分布的变化提出了一个问题。当输入分布到学习系统改变时，也就是说经历协变量移位，这个通常通过领域适应来处理。但是，协变量移位的概念可以作为一个整体延伸超出学习系统，适用于他自身的部分，比如子网络或者一个层。考虑一个网络的计算



其中和可以任意变换，参数学习去最小化损失。学习可以被视为如果输入如果输入被送入到子网络中



例如，一个梯度下降步骤



（批量大小为m学习率）完全等同于一个输入为的独立网络。因此，输入分布属性使得训练更有效—比如在训练和测试数据之间有相同的分布—也适用于子网络的训练。因此有利于的分布随时间保持不变。然后不必调整以补偿分布的变化。

固定一个子网络输入的分布将对子网络外的层产生积极的影响。用一个sigmoid激活函数考虑一个层，其中是层输入，权重矩阵和阈值向量是学习的层参数， 。随着增加，趋向于0.这意味着对于的所有维度，除了那些具有小的绝对值的，梯度流向下到会消失并且模型训练会变慢。但是因为是受，和下面所有层参数的影响，在训练期间对这些参数的改变将可能将的许多维度移动到非线性的饱和状态并且收敛减慢。这种效果是随着网络的深度的增加而放大。在实际应用中，饱和问题和导致的消失梯度通常通过使用Rectified Linear Units（ReLU）来解决。，仔细的初始化和小的学习率。然而，如果我们可以确保非线性输入的分布在网络训练时保持更加稳定，那么优化将不太可能在饱和状态中停滞，并且训练将加速。

我们将训练过程中深度网络内部节点分布的变化作为内部协变量转移，消除它可以提供一个更快的训练，对此我们提出了一个新的机制—批量归一化，它将减少内部协变量转移，这样做可以大大地加快深度神经网络的训练。它通过一个归一化步骤—固定层输入的平均值和方差不变来实现。通过减少梯度对参数规模或其初始值的依赖性，批量归一化还对网络的梯度流动具有有效的效果，这就允许我们在没有发散的风险下使用更高的学习率。此外，批量归一化正则化模型可以减少对Dropout的需求。最后，通过防止网络陷入饱和模式使得批量归一化可以使用饱和非线性。

在4.2节，我们将批量归一化运用到性能最佳的ImageNet分类网络，结果表明我们可以只使用7％的培训步骤去匹配其性能，并且可以进一步大幅度的超过其精确度。使用用批量归一化训练的这种网络集合，我们可以获得前5的误差率，增强了在ImageNet上已知的最佳结果。

2 减少内部协变量转移

我们定义内部协变量转移为由于训练期间网络参数的变化引起的网络激活的分布的变化。为了增强训练，我们要寻求减少内部协变量转移。通过固定层输入的分布作为训练过程，我们期待提高训练速度，一直以来众所周知如果层输入被白化，则网络训练就会更快地收敛。即线性转换到具有零均值和单位方差并且去相关。由于每层观察由下面层产生的输入，因此实现每层输入有相同白化将是有利的。通过白化每层输入，我们将采取步骤实现输入的固定分布，以消除内部协变量转移的不良影响。

我们可以考虑在每个训练步骤或者以一定间隔的白化激活，也可以通过直接修改网络或者根据网络激活值改变优化算法的参数。但是，如果这些修改与优化步骤穿插，则梯度下降步骤可以尝试以需要更新归一化的方式更新参数，这样可以减少梯度步的影响。比如，考虑一个层，输入是加上学习阈值，并且通过减去在训练数据上计算的激活的平均值来对结果进行归一化：其中，是训练集上值的集合，。如果一个梯度下降步骤忽略了对的依赖性，则它更新的值就是，其中。然后。因此，对b的更新和随后的归一化中的变化这两者的组合导致层的输出没有改变，因此也没有损失。随着训练继续，将无限增长，而损失则保持固定不变。如果归一化不仅中心而且缩放激活，这个问题可能变得更糟。我们在初始试验中观察到，当归一化参数在梯度下降步骤外计算时模型就会放大。

上述方法的问题是梯度下降优化没有考虑归一化发生的事实。为了解决这个问题，我们要确保对于任何参数值网络都会产生具有期望分布的激活。这样做能允许相应模型参数的损失梯度考虑归一化以及其对模型参数的依赖性。再次让为层输入，把它视为一个向量，是训练数据集上这些输入的集合。可以将归一化写成一个转换



这不仅取决于给定的训练样本，而取决于所有样本，如果是由另一层产生的，则中的每一个都取决于。对于反向传播（backpropagation），我们需要计算雅可比（Jacobians）

；

忽略后一项导致上述爆炸。在这个框架内，白化层输入代价是非常大的，它需要计算协方差矩阵和它的平方根倒数，去产生白化的激活，以及用于反向传播的这些转换的求导。这就促使我们去寻找一种替代方案，用微分去执行输入的归一化这样做不需要在每个参数更新之后分析整个训练集。

一些以前的方法在单个训练样本上或者在图像网络情况下，在一个给定位置的不同特征映射上使用统计计算。但是，丢弃激活的绝对量度会改变网络的表示能力。相对于整个训练数据的统计，我们想要通过在一个训练样本里归一化激活来保存网络中的信息。